USE OF NON-LOCAL MEANS FILTER TO DENOISE IMAGE CORRUPTED BY SALT AND PEPPER NOISE

Subhojit Sarker, Shalini Chowdhury, Samanwita Laha and Debika Dey

Department of Electronics & Communication Engineering,
Siliguri Institute Of Technology, Sukna, Darjeeling, India
{subhojitsarker, shln.chowdhury, samanwita30, debika.deyl}@gmail.com

Abstract.

In image processing, restoration and noise reduction is expected to improve the qualitative inspection of the image and the performance of quantitative image analysis techniques. Different approaches for image restoration have been proposed, each of which has its own advantages and limitation. This paper combines the adaptive median filtering technique and the non-local means filtering algorithm for image denoising salt and pepper noise to yield a considerably high PSNR. The Non-Local means filter assumes that an image contains an extensive amount of self-similarity and exploits it to denoise an image. The analysis of the statistical results is presented in simulations done using MATLAB to demonstrate the advantages and limitations of using the proposed method over other existing techniques.

Key words: Salt and pepper noise, Image denoising, Adaptive median Filter, Non-Local means filter.

1. Introduction

Most of the image-processing applications involve denoising as one of its most widely used concepts. The purpose of image denoising is to restore the original image details as much as possible by removing the unwanted noise. Digital image is susceptible to a variety of noise, which affects the image quality. One of such noise is salt and pepper noise which is generated by image sensor defect. Salt and pepper noise is caused by defective pixel in camera sensors and often found in digital transmission and storage. When an image is corrupted by salt and pepper noise, the pixel values may have any random value within the maximum and minimum values in the dynamic range [1]. Salt and Pepper noise is a special type of impulse noise. The probability density function (PDF) is

\[ P(z) = \begin{cases} 
  p_a, & z = a \\
  p_b, & z = b \\
  0, & otherwise 
\end{cases} \quad (1) \]
If neither of the probability is zero then the impulse noise resembles salt and pepper granules, distributed randomly over the image, hence the name. The removal of salt and pepper noise is generally approached using median type filters [2]. Previously Standard Median Filtering technique used to be considered as a robust technique in terms of noise attenuation and edge preservation. However in this method, when the noise variance is more than 0.5, some details and edges of the image are smashed [3]. An appropriate method of salt and pepper reduction is one which increases signal to noise ratio while preserving the edges and other fine details. To achieve this, an adaptive structure of the median filter was developed [4]. This adaptive median filter ensures that most of the impulse noise is detected even at a high noise level provided that the window size is large enough. This method too seized to yield low error at higher noise variance i.e., about 0.7 and blurring of image becomes prominent [5].

To obtain a lower MAE and higher PSNR, this paper proposes a two phase filtering technique. The noisy image is first subjected to a standard Adaptive Median Filter [6]. The filtered image is then denoised using Non-Local Means Filtering technique [7]. The NLM filter was introduced by Buades in 2005 [8]. This method of image denoising relies on the weighted average of all pixel intensities where the family of weights depends on the similarity between the pixels and the neighborhood of the pixel being processed. The proposed method outperforms the Standard Median Filter as well as Adaptive Median Filter in terms of several performance parameters. The numerical result obtained supports this claim.

The rest of the paper is organized as follows. The background is presented in Section II, which includes the techniques for Median filter, Adaptive Median Filter as well as the Non-Local Means Filtering. Then in Section III, the proposed method is described and finally Section IV and V, reports the simulation results and concluding remarks respectively.

2. Background

2.1 Standard Median Filter

The Standard Median Filter selects each pixel in the image and compares its value with the pixel values of its nearby neighbors in order to determine whether or not it is representative of its surroundings. If its value is unrepresentative of the surrounding pixels then its value is replaced with the median of those values. For calculating the median, all the pixel values from the surrounding neighborhood are first sorted into numerical order and then the pixel being considered is replaced with the middle pixel value, as illustrated in Figure. 1. Incase the neighborhood of the pixel under consideration has an even number of pixels, the average of the two middle pixel values is taken to calculate the median value [9] [10].
As illustrated in Figure 1, the central pixel value for a [3×3] square neighborhood is 150. This value is rather unrepresentative of the surrounding pixels and hence is replaced with the median value: 132. Larger neighborhoods may produce more severe smoothing.

2.2 Adaptive Median Filter

The standard median filter performs well as long as the spatial noise density of the salt and pepper noise is not large. The filter performance degrades when the spatial noise variance of the salt and pepper noise increases [3]. Further with larger image and as the size of the kernel increases, the details and the edges becomes obscured [11]. The standard median filter does not take into account the variation of image characteristics from one point to another. The behavior of adaptive filter changes based on statistical characteristic of the image inside the filter region defined by the $m \times n$ rectangular window $S_{xy}$ [11]. The adaptive median filter differs from other adaptive filter as the size of the rectangular window $S_{xy}$ is made to vary depending on

(a) $z_{\text{min}}$ = Minimum gray level value in $S_{xy}$
(b) $z_{\text{max}}$ = Maximum gray level value in $S_{xy}$
(c) $z_{\text{med}}$ = Medians of gray level in $S_{xy}$
(d) $z_{xy}$ = Gray levels at coordinate $(x,y)$
(e) $S_{\text{max}}$ = Maximum allowed size of $S_{xy}$ [12].

The flowchart of adaptive median filtering is based on two levels is shown in figure 2
The adaptive median filtering algorithm works in two levels, denoted by LEVEL1 and LEVEL2. The application of AMF provides three major purposes: to denoise images corrupted by salt and pepper (impulse) noise; to provide smoothing of non-impulsive noise, and also to reduce distortion caused by excessive thinning or thickening of object boundaries. The values $Z_{\text{min}}$ and $Z_{\text{max}}$ are considered statistically by the algorithm to be ‘impulse like’ noise components, even if these are not the lowest and highest possible pixel values in the image.

The purpose of LEVEL1 is to determine if the median filter output $Z_{\text{med}}$ is impulse output or not. If LEVEL1 does find an impulse output then that would cause it to branch to LEVEL2. Here, the algorithm then increases the size of the window and repeats LEVEL1 and continues until it finds a median value that is not an impulse or the maximum window size is reached, the algorithm returns the value of $Z_{xy}$.

Every time the algorithm outputs a value, the window $S_{xy}$ is moved to the next location in the image. The algorithm is then reinitialized and applied to the pixels in the new location. The median value can be updated iteratively using only the new pixels, thus reducing computational overhead.

Fig. 2. Flowchart of Adaptive median filter
2.3 Non-Local Means Filtering

The approach of Non-local means filter was introduced by Buades in 2005 [7] based on non-local averaging of all pixels in the image. The method was based on denoising an image corrupted by white gaussian noise with zero mean and variance.

The approach of Non Local Means filtering is based on estimating each pixel intensity from the information provided from the entire image and hence it exploits the redundancy caused due to the presence of similar patterns and features in the image. In this method, the restored gray value of each pixel is obtained by the weighted average of the gray values of all pixels in the image. The weight assigned is proportional to the similarity between the local neighborhood of the pixel under consideration and the neighborhood corresponding to other pixels in the image [8].

Given a discrete noisy image $v = \{v(i)\}$ for a pixel $I$ the estimated value of $\text{NL}[v](i)$ is computed as weighted average of all the pixels i.e.:

$$\text{NL}[v](i) = \sum_{j \in I} w(i, j) \cdot v(j)$$  \hspace{1cm} (2)

where the family of weights $\{w(i, j)\}$ depend on the similarity between the pixels $i$ and $j$.

The similarity between two pixels $i$ and $j$ depends on the similarity of the intensity gray level vectors $v(N_i)$ and $v(N_j)$, where $N_k$ denotes a square neighborhood of fixed size and centered at a pixel $k$. The similarity is measured as a decreasing function of the weighted Euclidean distance, $||v(N_i) - v(N_j)||^2_{2,a}$, where $a > 0$ is the standard deviation of the Gaussian kernel.

The pixels with a similar grey level neighborhood to $v(N_i)$ have larger weights in the average. These weights are defined as,

$$w(i, j) = \frac{1}{Z(i)} \cdot e^{-\frac{||v(N_i) - v(N_j)||^2_{2,a}}{h^2}}$$  \hspace{1cm} (3)

where $Z(i)$ is the normalizing constant and the parameter $h$ acts as a degree of filtering. It controls the decay of the exponential function and therefore the decay of the weights as a function of the Euclidean distances.

3. Proposed Method

The original test image is corrupted with simulated salt and pepper noise with different noise variance ranging from 0.1 to 0.9. In the proposed denoising approach, the noisy image is first applied to an adaptive median filter. The maximum allowed size of the window of the adaptive median filter is taken to be 5X5 for effective filtering. The choice of maximum allowed window size depends on the application but a reasonable value was computed by experimenting with various sizes of standard median filter.

In the second stage the resultant image is subjected to NL-means filtering technique. The block diagram of the proposed method is shown in figure 3 below.
The performance of the filter depends on,
(a) radio of search window (value taken 7)
(b) radio of similarity window (value taken 5)
(c) degree of filtering (taken equal to the value of noise variance divided by 5). The performance of the proposed technique is quantified by using various performance metrics such as, mean average error (MAE), mean square error (MSE), signal to mean square error (S/MSE), signal to noise ratio (SNR) and peak signal to noise ratio (PSNR).

4. Experimentation

4.1 Simulation

Intensive simulations were carried out using several monochrome images, from which a 256x256 “pout.tif” and “coins.png” image is chosen for demonstration. The test image is corrupted by fixed value salt and pepper noise with noise variance varying from 0.1 to 0.9.

In this section the output of the proposed technique is compared with different standard methods such as Median filters MF (3X3) and MF (5X5), Adaptive median filter AMF (7X7). The results are quantified using the following well defined parameters such as,

A) Mean Average Error (MAE) [13]

\[ MAE = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} |r_{ij} - x_{ij}| \]  \hspace{1cm} (4)

where \( r_{ij} \) and \( x_{ij} \) denote the pixel values of the restored image and the original image respectively and \( M \times N \) is the size of the image.

B) Mean Square Error (MSE)

\[ MSE = \frac{1}{MN} \sum_{i=1}^{M} \sum_{j=1}^{N} [I(i,j) - \hat{I}(i,j)]^2 \]  \hspace{1cm} (5)
Where M and N are the total number of pixels in the horizontal and the vertical dimensions of the image. \( I \) and \( \hat{I} \) denote the original and filtered image [14].

C) Signal to Mean Square Error (S/MSE) [15]

\[
S/MSE = 10 \log_{10} \frac{\sum_{i=1}^{K} S_i^2}{\sum_{i=1}^{K} (S_i - S)\text{'s}}^2
\]  

(6)

where \( S_i \) and \( S \) denote the pixel values of the restored image and original image respectively.

D) Peak Signal to Noise Ratio [14]

\[
PSNR = 10 \log_{10} \left( \frac{255^2}{MSE} \right)
\]  

(7)

E) Signal to Noise Ratio (SNR) [15]

\[
SNR = 10 \log_{10} \frac{\sum_{x_i \in \Omega^2 \left( v(x_i)^2 - \hat{v}(x_i)^2 \right)}}{\sum_{x_i \in \Omega^2 \left( v(x_i) - \hat{v}(x_i) \right)}^2}
\]  

(8)

where \( v(x_i) \) is the true value of pixel \( x_i \) and \( \hat{v}(x_i) \) is the restored value of pixel \( x_i \).

4.2 Denoising Performance

In this paper, salt and pepper noise was added to the original test images shown in Figure 4, with noise variance ranging from 0.1 to 0.9. The results for noisy and denoised images are shown in Figure 5 and the performance metrics obtained are shown in Table 1 for the ‘pout.tif’ image.

The performance of the proposed algorithm is tested for various levels of noise variance in ‘coins.png’ image and compared with standard filters namely Standard Median Filter (MF) of window size (3x3) and (5x5), Adaptive Median Filter (AMF), in terms of Mean Absolute Error (MAE) [13], Signal-To-Noise Ratio (SNR) [15], and the results are shown in Table 2.

![Fig. 4. Original test images (A) “pout.tif” image (B) “coins.png” image](image-url)
Fig. 5 (A-H). Noisy and denoised “pout.tif” image at noise variances 0.1, 0.3, 0.6 and 0.9 respectively.
Table 1: Performance matrices for noisy & denoised “pout.tif” image using proposed method

<table>
<thead>
<tr>
<th>VARIANCE</th>
<th>IMAGE TYPE</th>
<th>MAE</th>
<th>MSE</th>
<th>SMSE</th>
<th>SNR</th>
<th>PSNR</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>NOISY</td>
<td>0.0488</td>
<td>0.0331</td>
<td>5.7234</td>
<td>7.9367</td>
<td>62.9301</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.0158</td>
<td>0.0008</td>
<td>19.8079</td>
<td>22.8822</td>
<td>78.9154</td>
</tr>
<tr>
<td>0.2</td>
<td>NOISY</td>
<td>0.0971</td>
<td>0.0661</td>
<td>3.9275</td>
<td>5.6993</td>
<td>59.9302</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.0221</td>
<td>0.0012</td>
<td>18.0476</td>
<td>21.1428</td>
<td>77.1962</td>
</tr>
<tr>
<td>0.3</td>
<td>NOISY</td>
<td>0.1505</td>
<td>0.1033</td>
<td>3.0415</td>
<td>4.4873</td>
<td>57.9881</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.0290</td>
<td>0.0019</td>
<td>16.0876</td>
<td>19.1962</td>
<td>75.2624</td>
</tr>
<tr>
<td>0.4</td>
<td>NOISY</td>
<td>0.2026</td>
<td>0.1385</td>
<td>2.6022</td>
<td>3.8277</td>
<td>56.7149</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.0341</td>
<td>0.0026</td>
<td>14.6900</td>
<td>17.8313</td>
<td>73.9287</td>
</tr>
<tr>
<td>0.5</td>
<td>NOISY</td>
<td>0.2520</td>
<td>0.1712</td>
<td>2.3012</td>
<td>3.3829</td>
<td>55.7970</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.0376</td>
<td>0.0031</td>
<td>13.8464</td>
<td>17.0411</td>
<td>73.1881</td>
</tr>
<tr>
<td>0.6</td>
<td>NOISY</td>
<td>0.2982</td>
<td>0.2031</td>
<td>2.1173</td>
<td>3.0817</td>
<td>55.0538</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.0433</td>
<td>0.0051</td>
<td>11.7641</td>
<td>14.9160</td>
<td>71.0232</td>
</tr>
<tr>
<td>0.7</td>
<td>NOISY</td>
<td>0.3500</td>
<td>0.2382</td>
<td>1.9545</td>
<td>2.8186</td>
<td>54.3619</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.0495</td>
<td>0.0078</td>
<td>10.0867</td>
<td>13.1825</td>
<td>69.2365</td>
</tr>
<tr>
<td>0.8</td>
<td>NOISY</td>
<td>0.4046</td>
<td>0.2772</td>
<td>1.8425</td>
<td>2.6129</td>
<td>53.7032</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.0719</td>
<td>0.0205</td>
<td>6.7615</td>
<td>9.4248</td>
<td>65.0179</td>
</tr>
<tr>
<td>0.9</td>
<td>NOISY</td>
<td>0.4501</td>
<td>0.3072</td>
<td>1.7628</td>
<td>2.4757</td>
<td>53.2571</td>
</tr>
<tr>
<td></td>
<td>DENOISED</td>
<td>0.1532</td>
<td>0.0670</td>
<td>3.8486</td>
<td>5.6259</td>
<td>59.8678</td>
</tr>
</tbody>
</table>

Fig. 6. Graphical analysis of performance matrices for “pout.tif” image
The graphical analysis of the results obtained in Table 1 show that if the image corrupted by salt and pepper noise is of low variance, then the Mean Absolute Error (MAE) and the Mean Square Error (MSE) is comparatively low. However with the increase in noise variance there is a corresponding increase in calculated error. It can be seen that error in the denoised image has been reduced to a large extent as compared to the noisy image. There is an improvement in SNR and PSNR as well. For instance for noise variance of 0.7 there is an improvement of PSNR by about 15db.

The performances of the various denoising filters for “coins.png” are shown in Figure 7 and a comparative graphical analysis with respect to PSNR values is shown in Figure 8.

**Fig. 7.** (A) Original coins.png image (B) Noisy image with variance-0.6 (C) Median Filter (3x3) output (D) Median Filter (5x5) output (E) Adaptive median filter output (F) Denoised image using proposed method
Table 2: Comparison of PSNR values with other existing methods for a coins.png image

<table>
<thead>
<tr>
<th>Noise Variance</th>
<th>Noisy image</th>
<th>Median (3x3) Filter</th>
<th>Median (5x5) Filter</th>
<th>Adaptive median Filter</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>62.8843</td>
<td>72.8380</td>
<td>72.1363</td>
<td>75.5751</td>
<td>75.9079</td>
</tr>
<tr>
<td>0.2</td>
<td>59.8496</td>
<td>71.7523</td>
<td>71.1980</td>
<td>75.1047</td>
<td>75.1396</td>
</tr>
<tr>
<td>0.3</td>
<td>58.1739</td>
<td>68.6102</td>
<td>70.5042</td>
<td>73.8276</td>
<td>74.4453</td>
</tr>
<tr>
<td>0.4</td>
<td>56.7958</td>
<td>64.5924</td>
<td>68.6035</td>
<td>71.0273</td>
<td>72.1395</td>
</tr>
<tr>
<td>0.5</td>
<td>55.8869</td>
<td>61.8947</td>
<td>66.5301</td>
<td>70.0929</td>
<td>71.6597</td>
</tr>
<tr>
<td>0.6</td>
<td>55.3055</td>
<td>60.4782</td>
<td>65.2498</td>
<td>68.9711</td>
<td>70.4294</td>
</tr>
<tr>
<td>0.7</td>
<td>54.5339</td>
<td>57.5156</td>
<td>60.9605</td>
<td>65.0621</td>
<td>67.5890</td>
</tr>
<tr>
<td>0.8</td>
<td>53.7801</td>
<td>55.4299</td>
<td>57.1478</td>
<td>61.0267</td>
<td>64.3555</td>
</tr>
<tr>
<td>0.9</td>
<td>53.3445</td>
<td>54.1160</td>
<td>54.9722</td>
<td>56.7966</td>
<td>58.6945</td>
</tr>
</tbody>
</table>

From Table 2 it is evident that the Proposed Method performs best in terms of the peak signal-to-noise ratio (i.e PSNR). Experimental results obtained, show that at higher noise variance the proposed method restores the original image much better than standard non-linear median-based filter and adaptive median filter. For instance at noise variance of 0.7 the PSNR of the restored image improves by about 14db as compared to the noisy image as opposed to the case of 11db for AMF.

Fig. 8. Graph showing comparison of PSNR obtained using different filters for coins.png image
5. Conclusion

In this paper a new method is developed for restoration of an image, corrupted by salt and pepper noise. For lower values of noise variance, the existing filters like median filter and adaptive median filter can denoise salt and pepper noise, but fail to remove noise effectively as the noise variance increase. This paper proposes a method to handle salt and pepper noise even at higher variances.

In order to demonstrate the performance of the proposed method, extensive simulation experiments have been carried out on a variety of standard test images to compare the proposed method with many other existing techniques. Experimental results simulated with MATLAB 7 indicate that the proposed method performs significantly better than many other existing techniques when the noise variance is higher and these results are also graphically analyzed for comparative study.

Although the proposed method outperforms the existing denoising techniques at higher value of noise variances, but scope of improvement still exists. As this method involves a two stage process, the number of computations is very high and hence the simulation time increases with increase in the size of the corrupted image. So as a future work, modifications may be incorporated to reduce the computation time and make the algorithm faster.

References


